#14
5-19-04
DW

I hereby certify that this correspondence is being transmitted by facsimile to the U.S. Patent and Trademark Office at (703) 746 - 7238 APRIL 30, 2003
(Date of Deposit)

FRANK C. NICHOLAS (33,983)
Name of applicant, assignee or registered representative

Signature

APRIL 30, 2003
Date of Signature

PATENT
Case No. PHA 23,756
(7790/105)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re patent application of:

WINTHROP L. SAVILLE, ET AL.

Serial No.: 09/391,647

Filed: SEPTEMBER 7, 1999

Title: METHOD FOR FORMING VARIABLE LENGTH INSTRUCTINS IN A PROCESSING SYSTEM

Examiner: ELLIS, R.L.

Group Art Unit: 2183

APPEAL BRIEF

Honorable Commissioner of Patents and Trademarks
Washington, D.C. 20231

Dear Sir:

Appellants herewith respectfully present their Brief on Appeal as follows:

1. REAL PARTY IN INTEREST

The real party in interest is Assignee U.S. Philips Corporation, which is a Delaware corporation having an office and a place of business at 100 East 42nd Street, New York, NY 10017.

2. RELATED APPEALS AND INTERFERENCES

Appellants and the undersigned attorney are not aware of any other appeals or interferences which will directly affect or be directly affected by or having a bearing on the Board's decision in the pending appeal.

3. STATUS OF CLAIMS

Claims 14-26 are currently pending in the application and are the claims on appeal. See, the Appendix. Claims 14, 20, and 21 are the independent claims. Claims 15-19 depend from independent claim 14. Claims 22-26 depend from independent claims 20 and 21. Claims 14-26 stand finally rejected under 35 U.S.C. §102(b) as being anticipated by U.S. patent No. 4,454,578 to *Matsumoto* et al.

4. STATUS OF AMENDMENTS

A reply under 37 C.F.R. §1.112 involving a cancellation of claims 1-13, and an addition of claims 14-26 was filed on 06/20/02 and entered into the application. A reply under 37 C.F.R.

§1.116 involving an amendment of claims 20 and 21, and a cancellation of claim 22 was filed on 11/26/02, but not entered into the application.

5. SUMMARY OF THE INVENTION

Each instruction of the present invention includes an opcode portion and a parameter portion including one or more parameter bytes of a predetermined bit length (e.g.. 8 bits). The present invention discloses four (4) basic types of parameter portions. The first and second parameter portion types, as subsequently described herein, are prior art to the present invention. The third and fourth parameter portion types, as subsequently described herein, are new and unique to the present invention.

The first parameter portion type represents unsigned data that is not compressed whereby a total bit length m of the data equals a total bit length n of the parameter portion. An example of an instruction formed in accordance with the first parameter portion type including a parameter portion having a parameter byte 10011011 is illustrated in FIG. 3A of the present application, where $m = n = 8$ bits.

The second parameter portion type represents signed data that is not compressed whereby total bit length m of the data equals total bit length n of the parameter portion. An example of an instruction formed in accordance with the second parameter portion type including a parameter portion having a parameter byte 01101101 is illustrated in FIG. 3B of the present application, where $m = n = 8$ bits.

The third type is a parameter portion representing unsigned data that may or may not be compressed whereby total bit length m of the data equals total bit length n of the parameter portion if the data is not compressed, and whereby total bit length m of the data is greater than total bit length n of the parameter portion if the data is compressed. An expansion bit within a parameter byte indicates whether the data value bits of that particular byte should be expanded or data value bits from additional parameter bytes, if any, should be read.

An example of an instruction in accordance with the third parameter portion type including a parameter portion having a first parameter byte 01010101 and a second parameter byte 11110000 is illustrated in FIG. 3C of the present application, wherein n = 16 bits of the parameter portion. The expansion bit (e) in the first parameter byte is (0). Thus, the data value bits of the second parameter byte are read to obtain an overall data value of 0101010 111110000, where m = 16 bits of data length.

Another example of an instruction in accordance with the third parameter portion type including a first parameter portion having a parameter byte 11010101 and a second parameter portion having a parameter byte 11100110 is illustrated in FIG. 3D of the present application, wherein n = 8 bits for both parameter portions. The expansion bit (e) in the first parameter byte is (1). Thus, the data value bits of the first parameter byte is expanded for example to 00000000 01010101, where the first parameter byte has a data length of 16 bits. The expansion bit (e) in the second parameter byte is also (1). Thus, the data value bits of the second parameter byte is expanded for example to 00000000 01100110, where the second parameter byte has a data

length of 16 bits. The result is an overall data value of 00000000 01010101 00000000 01100110, where m = 32 bits of data length.

Another example of an instruction in accordance with the third parameter portion type including a parameter portion having a first parameter byte 01010101, a second parameter byte 01110000, a third parameter byte 01100110 and a fourth parameter byte 11110000 is illustrated in FIG. 3E of the present application, wherein n = 32 bits of the parameter portion. The expansion bit (e) in the first parameter byte is (0). Thus, the data value bits of the second parameter byte are read. The expansion bit (e) in the second parameter byte is (0). Thus, the data value bits of the third parameter byte are read. The expansion bit (e) in the third parameter byte is (0). Thus, the data value bits of the fourth parameter byte are read. The result is a data value of 01010101 011100000 1100110 11110000, where m = 32 bits of data length.

Another example of an instruction in accordance with the third parameter portion type including a parameter portion having a first parameter byte 01010101, a second parameter byte 01110000 and a third parameter byte 11100110 is illustrated in FIG. 3F of the present application, wherein n = 24 bits of the parameter portion. The expansion bit (e) in the first parameter byte is (0). Thus, the data value bits of the second parameter byte are read. The expansion bit (e) in the second parameter byte is (0). Thus, the data value bits of the third parameter byte are read. The expansion bit (e) in the third parameter byte is (1). Thus, the data value bits of the third parameter byte are expanded for example to 00000000 01100110. The result is a data value of 01010101 01110000 00000000 01100110, where m = 32 bits of data length.

Additional examples of instructions in accordance with the third parameter portion type are illustrated in FIGS. 3K and 3L of the present application.

The fourth type is a parameter portion representing signed data that may or may not be compressed whereby total bit length m of the data equals total bit length n of the parameter portion if the data is not compressed, and whereby total bit length m of the data is greater than total bit length n of the parameter portion if the data is compressed. An expansion bit within a parameter byte indicates whether the data value bits of that particular byte should be expanded or data value bits from additional parameter bytes, if any, should be read. Examples of instructions in accordance with the fourth type are illustrated in FIGS. 3G-3J of the present application.

6. <u>ISSUES</u>

Whether claims 14-19 and 22 are anticipated by *Matsumoto*.

7. <u>GROUPING OF CLAIMS</u>

The claims should be considered in two (2) separate groups:

Group I includes representative claim 14 and claims 15-19 depending from claim 14, all of which specify a method of forming instructions for execution in a processing system.

Group II is dependent claim 22, which specifies a method of forming instructions for execution in a processing system.

8. ARGUMENTS

Matsumoto. As illustrated in FIGS. 1A and 1B, *Matsumoto* discloses a short literal mode specifier and a long literal mode specifier, respectively, for an operand (i.e., parameter portions).

The short literal mode specifier includes a most significant bit ("MSB"), a TYPE field, a LENGTH field and a LITERAL VALUE field. A total bit length of the MSB is 1 bit. A total bit length of the TYPE field is 2 bits. A total bit length of the LENGTH field is 3 bits. A total bit length of the LITERAL VALUE field is either X or Y, where X equals 4 x the length indicated by the LENGTH field and Y equals 8 x the length indicated by the LENGTH field. *Matsumoto* teach the length of the LITERAL VALUE field ranges from a minimum of 4 bits (LENGTH field is 001, TYPE field indicates 4 bits) to a maximum of 56 bits (LENGTH field is 111, TYPE field indicates 8 bits). See, *Matsumoto* at column 2, lines 58-66.

The long literal mode specifier includes a most significant bit ("MSB"), a TYPE field, a 000 field, a LENGTH field and a LITERAL VALUE field. A total bit length of the MSB is 1 bit. A total bit length of the TYPE field is 2 bits. A total bit length of the 000 field is 3. A total bit length of the LENGTH field of is 8 bits. A total bit length of the LITERAL VALUE field is either X or Y, where X equals 4 x the length indicated by the LENGTH field and Y equals 8 x the length indicated by the LENGTH field. *Matsumoto* teach the length of the LITERAL VALUE field ranges from a minimum of 4 bits (LENGTH field is 00000001, TYPE field indicates 4 bits) to a maximum of 2,040 bits (LENGTH field is 11111111, TYPE field indicates 8 bits). See, *Matsumoto* at column 2, line 55 to column 3, line 2.

Matsumoto further discloses an instruction aligner ALIG 402 (FIG. 4A) for aligning an operand specifier 11A (FIG. 4A) bit by bit in a sequence 14A (FIG. 4B) to be decoded by an operand specifier decoder OS-DEC 505 (FIG. 4B). <u>See</u>, *Matsumoto* at column 6, lines 31-35; and column 7, lines 10 and 11. *Matsumoto* fails to teach or suggest a compression of the LITERAL VALUE fields of the specifiers whereby the LITERAL VALUE fields would have to be expanded during a decoding of the operand specifiers. Thus, *Matsumoto* fails to disclose, teach or suggest a need for an expansion bit in the operand specifier wherein the expansion bit indicates whether the data bits of the LITERAL VALUE field should be expanded by decoder 505.

Moreover, *Matsumoto* teaches the short literal mode specifier and the long literal mode specifier are two separate and distinct operand specifiers. Specifically, *Matsumoto* teaches the long literal mode specifier is to be used whenever a length of the literal data needs to be greater than the 56 bit maximum of the short literal mode specifier. <u>See</u>, *Matsumoto* at column 2, line 55 to column 3, line 2. Thus, *Matsumoto* can not be interpreted as proposed by Examiner Ellis that the 000 field of the long literal mode specifier indicates an expansion of the short literal mode specifier, because a substitution of the long literal mode specifier for the short literal mode specifier whenever the address is greater than 56 bits negates any purpose for expanding the short literal mode specifier.

<u>Group 1.</u> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed.

Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). Thus, to warrant the §102(b) rejection of independent claim 14, *Matsumoto* must disclose each and every limitation of independent claim 14 in as complete detail as is contained in independent claim 14. The Appellants respectfully assert that Matsumoto clearly fails to disclose, teach or suggest "providing a first parameter byte including . . . a first expansion bit indicative of whether the processing system expands the first set of data value bits or reads any additional parameter bytes including additional sets of data value bits" as recited in independent claim 14. Withdrawal of the rejection of independent claim 14 under 35 U.S.C. §102(b) as being anticipated by *Matsumoto* is therefore respectfully requested.

Claims 15-19 depend from independent claim 14. Therefore, dependent claims 15-19 include all of the elements and limitations of independent claim 14. It is therefore respectfully submitted by the Appellants that dependent claims 15-19 are allowable over *Matsumoto* for at least the same reason as set forth above with respect to independent claim 14. Withdrawal of the rejection of dependent claims 15-19 under 35 U.S.C. §102(b) as being anticipated by *Matsumoto* is therefore respectfully requested.

Group II. To warrant the §102(b) rejection of dependent claim 22, *Matsumoto* must disclose each and every limitation of dependent claim 22 in as complete detail as is contained in dependent claim 22. The Appellants respectfully assert that Matsumoto clearly fails to disclose, teach or suggest "wherein the parameter portions further includes a second indicator representative of whether to expand the plurality of data value bits" as recited in independent

claim 22. Withdrawal of the rejection of dependent claim 22 under 35 U.S.C. §102(b) as being anticipated by *Matsumoto* is therefore respectfully requested.

Dated: **April 30, 2003**

Respectfully submitted,
WINTHROP L. SAVILLE et al.

U.S. PHILIPS
580 White Plains Road
Tarrytown, New York 10591
Phone: (914) 333-9606
Fax: (914) 332-0615

Robert J. Kraus
Registration No. 26,358
Attorney for Appellants

CARDINAL LAW GROUP
Suite 2000
1603 Orrington Avenue
Evanston, Illinois 60201
Phone: (847) 905-7111
Fax: (847) 905-7113

Frank C. Nicholas
Registration No. 33,983
Attorney for Appellants

APPENDIX

14. A method of forming instructions for execution in a processing system, said method comprising:

providing an opcode portion determining at least one instruction to be performed by the processing system; and

providing a first parameter byte including a first set of data value bits, and a first expansion bit indicative of whether the processing system expands the first set of data value bits or reads any additional parameter bytes including additional sets of data value bits.

15. The method of claim 14, wherein said first byte further has a sign bit indicative of whether the first set of data value bits represents a positive number or a negative number.

16. The method of claim 14, further comprising

providing a second parameter byte including a second set of data value bits, and a second expansion bit indicative of whether the processing system expands the second set of data value bits or reads any additional parameter bytes including additional sets of data value bits.

17. The method of claim 16, wherein said first byte further has a sign bit indicative of whether the first set of data value bits and the second set of data value bits collectively represent a positive number or a negative number.

18. The method of claim 16, further comprising

providing a third parameter byte including a third set of data value bits, and a third expansion bit indicative of whether the processing system expands the third set of data value bits or reads any additional parameter bytes including additional sets of data value bits.

19. The method of claim 18, wherein said first byte further has a sign bit indicative of whether the first set of data value bits, the second set of data value bits and the third set of data value bits collectively represent a positive number or a negative number.

20. A method of forming instructions for execution in a processing system, said method comprising:

providing an opcode portion determining at least one instruction to be performed by the processing system; and

providing a parameter portion including a plurality of data value bits, and a first indicator representative of a number of the plurality of data value bits.

21. A method of forming instructions for execution in a processing system, said method comprising:

providing an opcode portion determining at least one instruction to be performed by the processing system; and

providing a parameter portion including a plurality of data value bits, and a first indicator representative a number of bytes in the parameter portion.

22. The method of claim 20 or 21, wherein the parameter portions further includes a second indicator representative of whether to expand the plurality of data value bits.

23. The method of claim 20 or 21, wherein the parameter portions further includes a second indicator representative of whether the plurality of data value bits represent a positive number or a negative number

24. The method of claim 20 or 21, wherein the opcode portion defines a number of parameters in the parameter portion.

25. The method of claim 20 or 21, wherein the opcode portion defines an uncompressed length of the plurality of data value bits.

26. The method of claim 20 or 21,

wherein the parameter portion includes a plurality of parameter bytes; and

wherein the opcode portion determines an order of arrangement of the plurality of parameter bytes.